计算机网络安全大作业技术报告(防火墙课题)

张金源-76066001

计算机科学与技术, 北京航空航天大学, 北京, 中国
michael.201096@gmail.com

摘要 防火墙指的是由软件和硬件设备组合而成、在内部网和外部网之间、专用网和公共网之间的边界上构造的保护屏障，是一种获取安全性方法的形象说法。防火墙使Internet之间建立起一个安全网关，从而保护内部网免受非法用户的侵入。防火墙设置在网络的边界上，对于流经防火墙的包裹进行鉴别和过滤，防火墙还可以根据特定协议所使用的指定端口号进行过滤。在Linux Ubuntu操作系统环境下，将Netfilter模块加载至内核。最终，分别实现基于协议的、基于IP地址的、基于端口号的防火墙构建方法。

Keywords: 防火墙，计算机网络安全，Netfilter

1 Introduction

1.1 防火墙背景与意义

防火墙指的是由软件和硬件设备组合而成、在内部网和外部网之间、专用网和公共网之间的边界上构造的保护屏障，是一种获取安全性方法的形象说法。防火墙使Internet之间建立起一个安全网关，从而保护内部网免受非法用户的侵入 [**?**]。防火墙主要由服务访问规则、验证工具、包过滤和应用网关4个部分组成。防火墙就是一个位于计算机和它所连接的网络之间的软件或硬件。该计算机流入流出的所有网络通信和数据包均要经过此防火墙。防火墙能够强化安全策略，有效地记录Internet上的活动。并且防火墙有效隔绝了内网与外网，防止内部问题外泄，也防止外部威胁入内。

1.2 相关技术分析

防火墙设置在网络的边界上，对于流经防火墙的包裹进行鉴别和过滤。所有防火墙都具有IP地址过滤功能，检查IP包头，根据其IP源地址和目标地址做出放行/丢弃决定。防火墙还可以根据特定协议所使用的指定端口号进行过滤。

1.3 主要研究内容

在Linux Ubuntu操作系统环境下，将Netfilter模块加载至内核。最终，分别实现基于协议的、基于IP地址的、基于端口号的防火墙构建方法。

2 实验目的及原理

2.1 实验目的

1. 理解防火墙技术的基本工作原理；
2. 理解Linux环境中Netfilter/IPTables的工作机制；
3. 掌握对Netfilter内核模块进行扩展编程的基本方法；
4. 掌握通过IPTables构建防火墙的基本方法。

2.2 实验要求

1. 对Netfilter内核模块进行扩展编程来实现简单的防火墙；
2. 实现基于协议的数据报过滤功能；
3. 实现基于源IP地址的数据报过滤功能；
4. 实现基于目的端口的TCP包过滤功能。

2.3 实验内容及原理

学习Netfilter和内核模块的相关知识，做三个分别基于协议、IP、端口的内核Netfilter模块。加载内核模块，测试防火墙的功能（丢弃某个协议、端口、IP的数据包）。基于Netfilter的防火墙包含4个文件， filter_ip.c、filter_port.c、 filter_prot.c、 makefile。其中filter_ip.c中代码实现的功能就是基于源IP地址过滤，filter_port.c实现了基于目的端口过滤，filter_prot.c实现了基于协议过滤的功能。linux内核编程中的三个函数，初始化、钩子函数、清除模块函数。

3 实验环境及组网图

本实验环境为Linux Ubuntu 14.04 LTS，然后因为设备数量有限的问题于是在本实验我用虚拟机（VMware），Ubuntu 14.04将在虚拟机下安装然后采用桥接方式将主机和虚拟机连接。实验组网（如图 1所示）

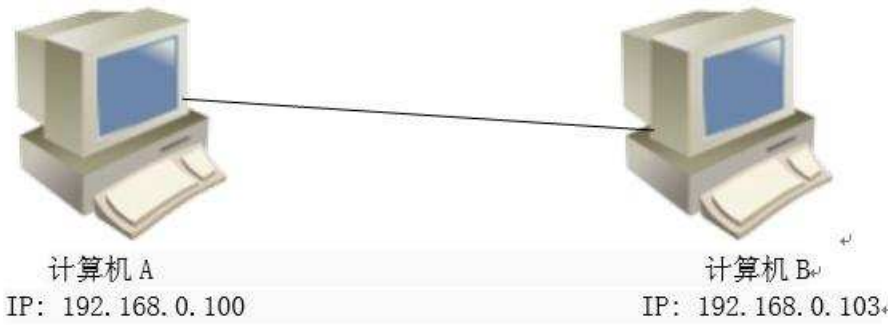

图 1. 实验组网图

4 实验步骤

在本实验我们做了三个实验，基于源IP地址过滤的防火墙、基于协议过滤的防火墙、基于端口的防火墙。

4.1 基于源IP地址过滤的防火墙

步骤1 在计算机A的命令行中开启root权限，否则原始套接字创建会失败。使用命令：su，进入root模式。

步骤2 编译源代码。命令行模式下，切换到代码所在目录，输入make后回车即可，编译后将生成三个文件，filter_ip.ko，filter_port.ko以及 filter_prot.ko。

步骤3 description将相关模块加载到内核。使用命令insmod ./filter_ip.ko将基于源IP地址的过滤模块加载到内核。

步骤4 通过计算机B ping 计算机A，观察 ping 命令结果。

步骤5 在计算机A中，通过dmesg命令查看系统内核日志结果。

4.2 基于协议过滤的防火墙

步骤1 在计算机A中，首先移除上个实验中加载的内核模块。使用命令rmmod ./filter_ip.ko卸载基于源IP地址的过滤模块。

步骤2 加载基于协议过滤的内核模块。使用命令insmod ./filter_prot.ko，加载基于协议过滤的内核模块。

步骤3 通过计算机A的浏览器浏览网页，观察结果，并通过dmesg命令查看系统内核日志结果。

步骤4 通过计算机B ping 计算机A，观察 ping 命令结果。

步骤5 在计算机A中，通过dmesg命令查看系统内核日志结果。

4.3 基于端口的防火墙

步骤1 在计算机A中，首先移除上个实验中加载的内核模块。使用命令rmmod ./filter_prot.ko卸载基于协议过滤的内核模块。

步骤2 加载基于端口过滤的内核模块。使用命令insmod ./filter_port.ko，加载基于端口过滤的内核模块。

步骤3 通过计算机A的浏览器浏览网页，观察结果，并通过dmesg命令查看系统内核日志结果。

步骤4 开启计算机A的telnet服务。使用netstat -a — grep telnet命令查看telnet运行状态，如果输出为空，表示没有开启该服务，按下面步骤进行配置，否则跳转到步骤9。

步骤5 安装openbsd-inetd。使用命令： sudo apt-get install openbsd-inetd

步骤6 安装telnetd。使用命令： sudo apt-get install telnetd 安装完成后，查看/etc/inetd.conf的内容，使用命令： cat/etc/inetd.conf — grep telnet 可以看到输出结果： telnet stream tcp nowait telnetd /usr/sbin/tcpd /usr/sbin/in.telnetd。

步骤7 重启openbsd-inetd。使用命令： sudo /etc/init.d/openbsd-inetd restart

步骤8 再次查看telnet运行状态。使用命令： netstat -a — grep telnet 输出结果为： tcp 0 0 *:telnet *:* LISTEN

步骤9 通过计算机B telnet 计算机A。在本实验组网中，应使用命令：telnet 192.168.0.103。观察telnet 结果。

步骤10 在计算机A中，通过dmesg命令查看系统内核日志结果。

5 实验结果

5.1 基于源IP地址过滤的防火墙

本实验结果如图 2所示，虚拟机（Ubuntu）将主机的IP进行过滤，这导致对主机的IP，当PING虚拟机时数据报文的结果都是Drop，这说明从主机IP的报文进不去虚拟机（Ubuntu）。

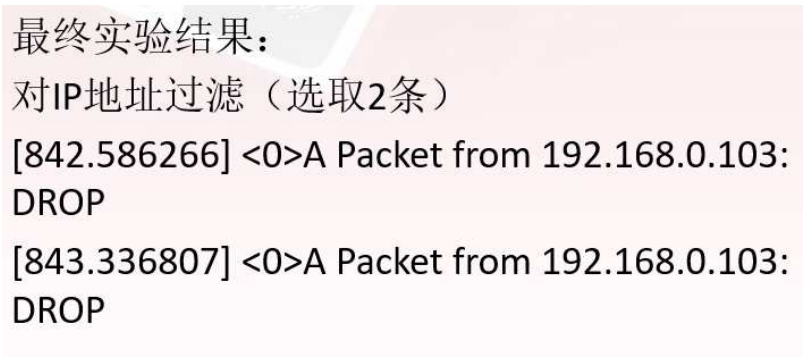

图 2. 实验1结果

5.2 基于协议过滤的防火墙

图 3. 实验2结果

本实验结果如图 5.2所示，将ICMP报文进行过滤，TCP和UDP报文都能正常通过（accept）但ICMP报文虚拟机没有接受(Drop)。在本实验说明防火墙在虚拟机拒绝了从主机A的ICMP报文。

5.3 基于端口的防火墙

本实验结果 4所示，将23端口行过滤，如果数据包从23端口发送的，防火墙将数据包拒绝（Drop）但如从别的端口防火墙接受（Accept）。

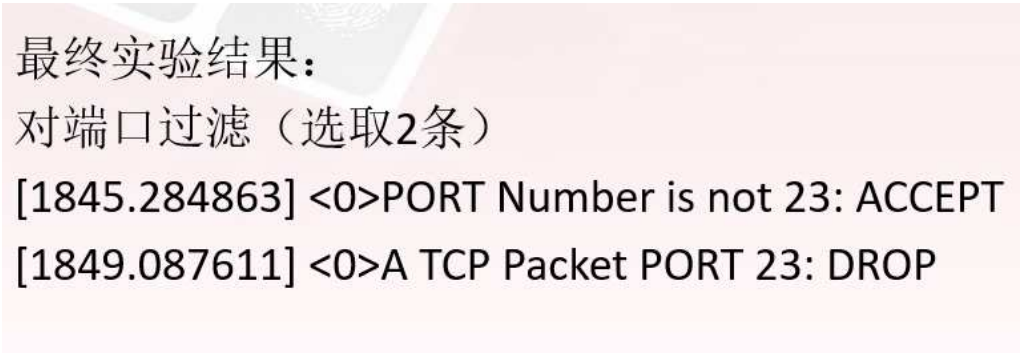

图 4. 实验1结果

6 总结

通过这门课的大作业我个人更理解防火墙的工作原理，而且我对信息安全有了新的知识了。在实验过程中，我们只遇到问题在实验设备的数量，为了解决这问题我们使用虚拟机解决这个问题，并将虚拟机与主机采用桥接方式连接。很感谢老师提供的参考文件，对我来说这些参考文件很有帮助，而这大作业也让我体会防火墙的工作原理。

7 参考资料

– 百度百科-防火墙
– 课程中心-大作业参考资料-实验一 Netfilter实验（2019年修订）